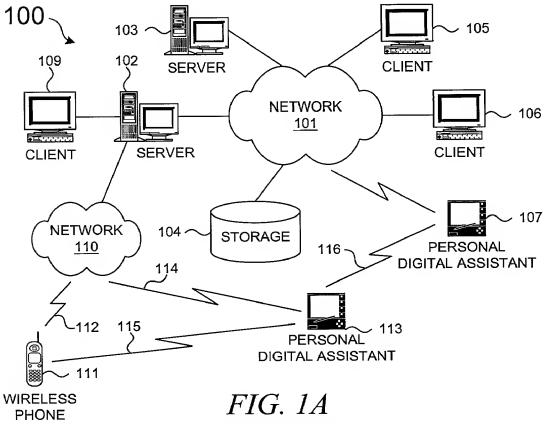

FIG. 1A
(PRIOR ART)

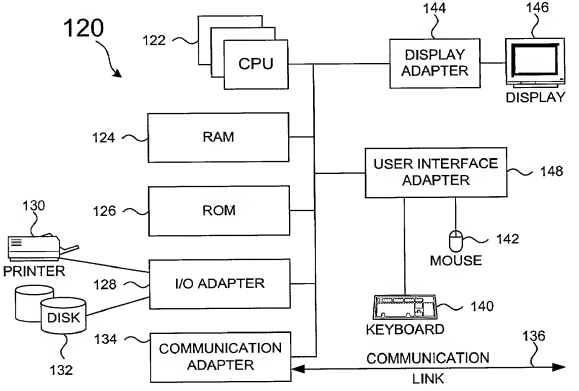

FIG. 1B
(PRIOR ART)

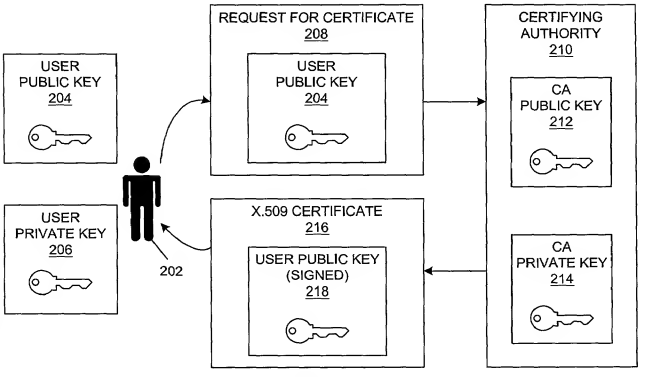

FIG. 2
(PRIOR ART)

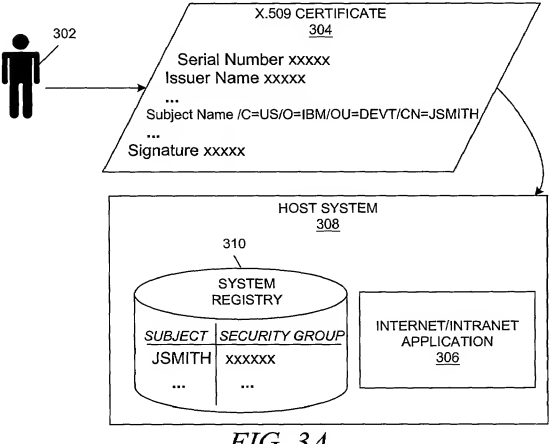

FIG. 3A
(PRIOR ART)

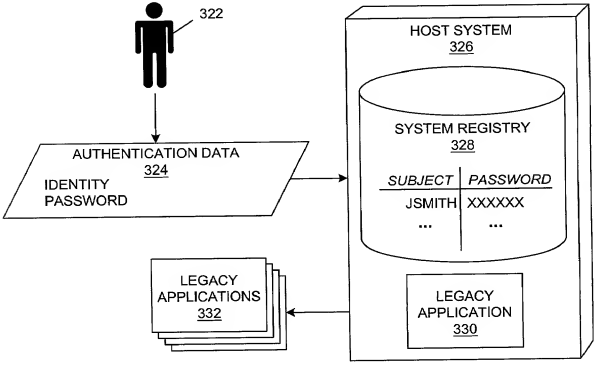

FIG. 3B
(PRIOR ART)

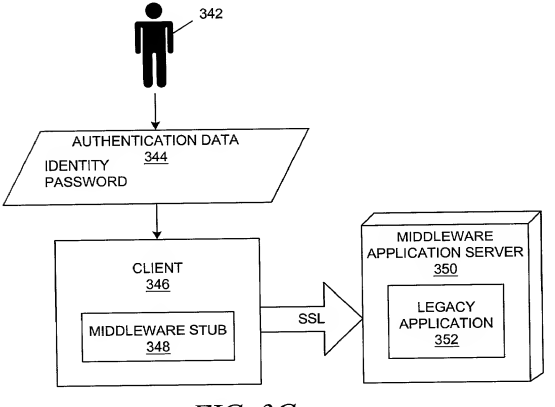

FIG. 3C
(PRIOR ART)

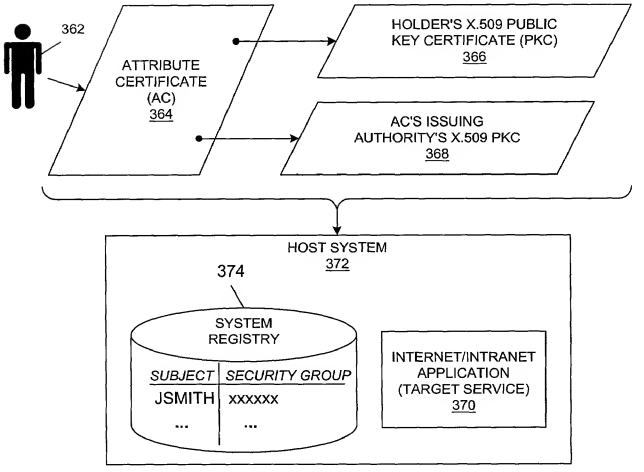

FIG. 3D
(PRIOR ART)

```
Certificate  ::=  SEQUENCE  {
      tbsCertificate              TBSCertificate,
      signatureAlgorithm          AlgorithmIdentifier,
      signature                   BIT STRING  }

TBSCertificate  ::=  SEQUENCE  {
      version              [0]    Version DEFAULT v1,
      serialNumber                CertificateSerialNumber,
      signature                   AlgorithmIdentifier,
      issuer                      Name,
      validity                    Validity,
      subject                     Name,
      subjectPublicKeyInfo        SubjectPublicKeyInfo,
      issuerUniqueID       [1]    IMPLICIT UniqueIdentifier OPTIONAL,
      subjectUniqueID      [2]    IMPLICIT UniqueIdentifier OPTIONAL,
      extensions           [3]    Extensions OPTIONAL    }

Version  ::=  INTEGER  {  v1(0), v2(1), v3(2)  }

CertificateSerialNumber  ::=  INTEGER

Validity ::= SEQUENCE {
      notBefore                   Time,
      notAfter                    Time }

Time ::= CHOICE {
      utcTime                     UTCTime,
      generalTime                 GeneralizedTime }

UniqueIdentifier  ::=  BIT STRING

SubjectPublicKeyInfo  ::=  SEQUENCE  {
      algorithm                   AlgorithmIdentifier,
      subjectPublicKey            BIT STRING  }

Extensions  ::=  SEQUENCE SIZE (1..MAX) OF Extension

Extension  ::=  SEQUENCE  {
      extnID                      OBJECT IDENTIFIER,
      critical                    BOOLEAN DEFAULT FALSE,
      extnValue                   OCTET STRING  }
```

FIG. 4A
(PRIOR ART)

```
AttributeCertificate ::= SEQUENCE {
      acinfo                AttributeCertificateInfo,
      signatureAlgorithm    AlgorithmIdentifier,
      signatureValue        BIT STRING
}

AttributeCertificateInfo ::= SEQUENCE {
      version                   AttCertVersion DEFAULT v1,
      holder                    Holder,
      issuer                    AttCertIssuer,
      signature                 AlgorithmIdentifier,
      serialNumber              CertificateSerialNumber,
      attrCertValidityPeriod    AttCertValidityPeriod,
      attributes                SEQUENCE OF Attribute,
      issuerUniqueID            UniqueIdentifier OPTIONAL,
      extensions                Extensions OPTIONAL
}

AttCertVersion ::= INTEGER { v1(0), v2(1) }

Holder ::= SEQUENCE {
      baseCertificateID   [0] IssuerSerial OPTIONAL,
                          -- the issuer and serial number of
                          -- the holder's Public Key Certificate
      entityName          [1] GeneralNames OPTIONAL,
                          -- the name of the claimant or role
      objectDigestInfo    [2] ObjectDigestInfo OPTIONAL
                          -- if present, version must be v2
}

ObjectDigestInfo ::= SEQUENCE {
      digestedObjectType  ENUMERATED {
      publicKey            (0),
      publicKeyCert        (1),
      otherObjectTypes     (2) },
                                -- otherObjectTypes MUST NOT
                                -- be used in this profile
      otherObjectTypeID   OBJECT IDENTIFIER OPTIONAL,
      digestAlgorithm     AlgorithmIdentifier,
      objectDigest        BIT STRING
}
```

FIG. 4B
(PRIOR ART)

```
AttCertIssuer ::= CHOICE {
    v1Form              GeneralNames,  -- v1 or v2
    v2Form        [0]   V2Form    -- v2 only
}

V2Form ::= SEQUENCE {
    issuerName              GeneralNames  OPTIONAL,
    baseCertificateID  [0]  IssuerSerial  OPTIONAL,
    objectDigestInfo   [1]  ObjectDigestInfo  OPTIONAL
                            -- at least one of issuerName, baseCertificateID
                            -- or objectDigestInfo MUST be present}

IssuerSerial  ::=  SEQUENCE {
    issuer      GeneralNames,
    serial      CertificateSerialNumber,
    issuerUID   UniqueIdentifier OPTIONAL
}

AttCertValidityPeriod  ::= SEQUENCE {
    notBeforeTime  GeneralizedTime,
    notAfterTime   GeneralizedTime
}

Attribute ::= SEQUENCE {
    type        AttributeType,
    values      SET OF AttributeValue
                -- at least one value is required
}

AttributeType ::= OBJECT IDENTIFIER

AttributeValue ::= ANY DEFINED BY AttributeType
```

FIG. 4C
(PRIOR ART)

```
name        id-aca-authenticationInfo
OID         { id-aca 1 }
Syntax      SvceAuthInfo
values:     Multiple allowed

SvceAuthInfo ::=   SEQUENCE {
    service     GeneralName,
    ident       GeneralName,
    authInfo    OCTET STRING OPTIONAL
}
```

FIG. 4D
(PRIOR ART)

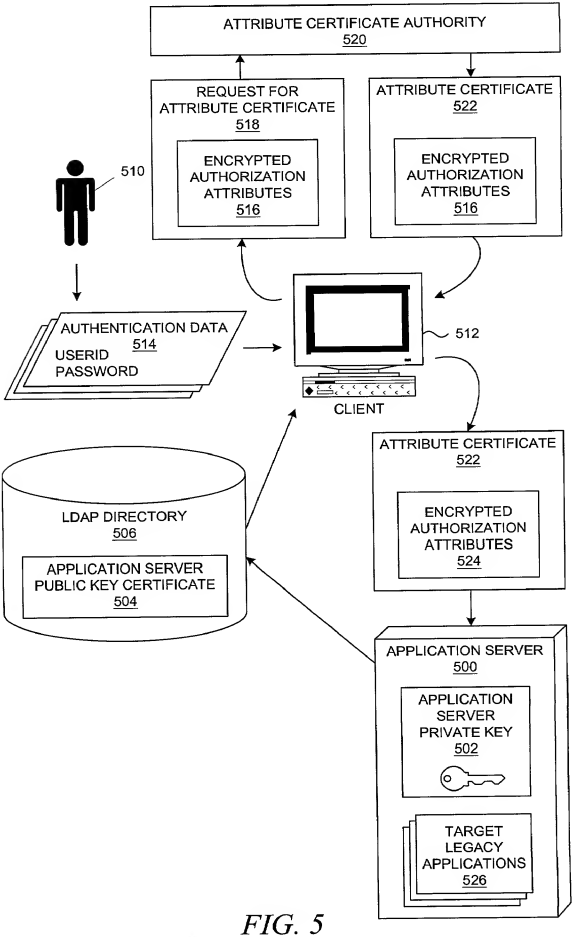
ATTRIBUTE CERTIFICATE AUTHORITY
520
REQUEST FOR ATTRIBUTE CERTIFICATE
518
ENCRYPTED AUTHORIZATION ATTRIBUTES
516
ATTRIBUTE CERTIFICATE
522
ENCRYPTED AUTHORIZATION ATTRIBUTES
516
510
AUTHENTICATION DATA
514
USERID
PASSWORD
512
CLIENT
ATTRIBUTE CERTIFICATE
522
ENCRYPTED AUTHORIZATION ATTRIBUTES
524
LDAP DIRECTORY
506
APPLICATION SERVER PUBLIC KEY CERTIFICATE
504
APPLICATION SERVER
500
APPLICATION SERVER PRIVATE KEY
502
TARGET LEGACY APPLICATIONS
526

FIG. 5

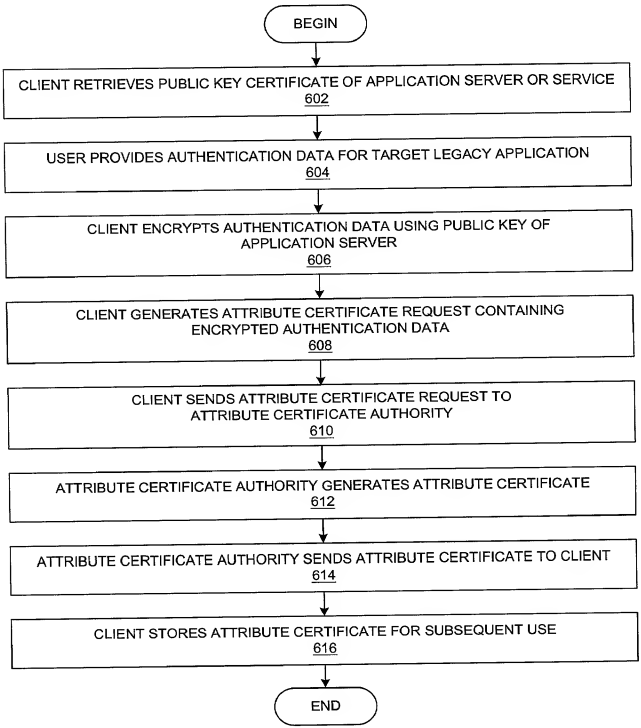
BEGIN
CLIENT RETRIEVES PUBLIC KEY CERTIFICATE OF APPLICATION SERVER OR SERVICE
602
USER PROVIDES AUTHENTICATION DATA FOR TARGET LEGACY APPLICATION
604
CLIENT ENCRYPTS AUTHENTICATION DATA USING PUBLIC KEY OF APPLICATION SERVER
606
CLIENT GENERATES ATTRIBUTE CERTIFICATE REQUEST CONTAINING ENCRYPTED AUTHENTICATION DATA
608
CLIENT SENDS ATTRIBUTE CERTIFICATE REQUEST TO ATTRIBUTE CERTIFICATE AUTHORITY
610
ATTRIBUTE CERTIFICATE AUTHORITY GENERATES ATTRIBUTE CERTIFICATE
612
ATTRIBUTE CERTIFICATE AUTHORITY SENDS ATTRIBUTE CERTIFICATE TO CLIENT
614
CLIENT STORES ATTRIBUTE CERTIFICATE FOR SUBSEQUENT USE
616
END

FIG. 6

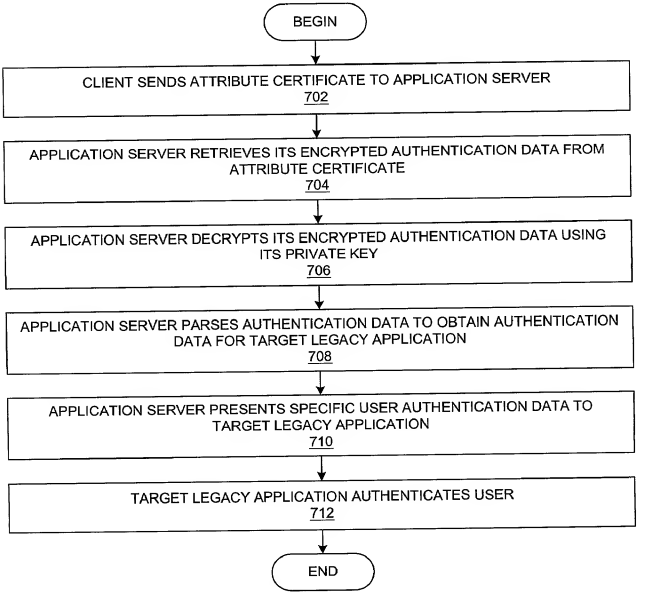
BEGIN
CLIENT SENDS ATTRIBUTE CERTIFICATE TO APPLICATION SERVER
702
APPLICATION SERVER RETRIEVES ITS ENCRYPTED AUTHENTICATION DATA FROM ATTRIBUTE CERTIFICATE
704
APPLICATION SERVER DECRYPTS ITS ENCRYPTED AUTHENTICATION DATA USING ITS PRIVATE KEY
706
APPLICATION SERVER PARSES AUTHENTICATION DATA TO OBTAIN AUTHENTICATION DATA FOR TARGET LEGACY APPLICATION
708
APPLICATION SERVER PRESENTS SPECIFIC USER AUTHENTICATION DATA TO TARGET LEGACY APPLICATION
710
TARGET LEGACY APPLICATION AUTHENTICATES USER
712
END

FIG. 7